Cassandra Crossing/ Storie di aerei caduti e del loro software

(437) Perché i piloti dei Boeing 737 MAX precipitati non sono riusciti a controllare i loro aerei?

Cassandra Crossing/ Storie di aerei caduti e del loro software

Figure 1: Alcuni dei 737MAX parcheggiati in Cina durante lo stop forzato - di Windmemories—Opera propria, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=77695741

(437) Perché i piloti dei Boeing 737 MAX precipitati non sono riusciti a controllare i loro aerei?

16 marzo 2019—Fiumi di inchiostro e di bit ci hanno giustamente inondato a seguito dei due disastri aerei che hanno coinvolto aeromobili Boeing 737 MAX, aerei di concezione modernissima e praticamente nuovi di zecca.

E proprio a seguito di questo eccesso mediatico è possibile che la maggior parte dei lettori si sia perso il motivo principale delle catastrofi, ormai assodato per la prima, altamente probabile per la seconda.

Poiché è un problema di ingegneria del software, di interazione uomo-macchina e di documentazione tecnica, è molto interessante descriverlo nella maniera più breve e concisa possibile.

Tre premesse tecniche:

1) le ali di un aereo spingono verso l'alto, e permettono all'aereo di volare, solo a partire da una certa velocità; se un aereo che sta volando tranquillamente scende sotto questa velocità cade in

verticale, improvvisamente e senza nessun preavviso, come un ferro da stiro. Questo fenomeno, incubo di qualsiasi pilota, si chiama “stallo”, e la velocità a cui questo avviene “velocità di stallo”;

2) la velocità di cui si parla è quella relativa all'aria, non al suolo, quindi è influenzata da eventi imprevedibili come un cambio di velocità del vento, e non è misurabile da strumenti come il GPS od il radar, che misurano invece la velocità assoluta rispetto al terreno;

3) per questo motivo esistono sensori multipli che, misurando la velocità dell'aria in vari punti dell'aereo, scattano in maniera “rumorosa” appena ci si avvicina alla velocità di stallo.

Ora, visto che gli aerei (e non solo) sono ormai comandati dal software, ed i comandi dei piloti sono solo uno degli input che il software analizza, come si deve realizzare un avviso di stallo imminente nell'abitacolo del pilota con qualcosa che sia meno rozzo di un sirena assordante ed una luce rossa lampeggiante?
Ma è semplice, integrandolo nell'interfaccia utente del pilota, già molto complessa, ed asservendolo al software di bordo.

E qui, pare, è nato il “problema”.

Nota: quanto segue è il miglior riassunto possibile elaborato dall'autore senza avere accesso diretto alla documentazione tecnica ed ai carteggi intercorsi tra Boeing e IATA. In quanto segue potrebbero quindi esserci imprecisioni, incompletezze od errori, di cui eventualmente l'autore si scusa e che correggerà ove esistenti.

Tutti sanno che negli aeromobili esiste il “pilota automatico”, che permette all'aereo di volare, senza l'interazione del pilota, sulla base di parametri preimpostati.
Quando il pilota ha il controllo manuale dell'aeromobile, questo automatismo è disinserito.

Ma visto che è sempre il software che controlla l'aereo, è più preciso riassumere affermando che il software ignora gli input dei comandi quando è attiva la modalità “pilota automatico”, ed invece li elabora e gli da la precedenza quando è attiva la modalità “volo manuale”.

Nel Boeing 737 MAX questo allarme di stallo si è evoluto in un sottosistema software con un nome preciso “Maneuver Characteristics Augmentation System”—MCAS, che è stato integrato nel software di bordo in questo modo:

1) in modalità “pilota automatico” MCAS partecipa a determinare le azioni del software di bordo come input supplementare.

2) in “volo manuale” invece MCAS non è disinserito, e nemmeno attivo solo come allarme, ma mantiene la sua funzione automatica, senza che di questo il pilota abbia particolare evidenza.

E pur trattandosi di una novità, sembra che questo fatto non fosse nemmeno spiegato a sufficienza nel manuale di aggiornamento, destinato ai piloti che avessero già esperienza di volo su modelli simili di aereo e dovessero essere addestrati al volo sul nuovo modello.

MCAS, pare ormai assodato, non è stato in grado di reagire bene al malfunzionamento di uno dei sensori di velocità dell'aria di cui era dotato.
Pare che MCAS non abbia rilevato il malfunzionamento di un singolo sensore, e che quindi abbia “pensato” che l'aereo stesse per entrare in condizioni di stallo.

Cosa deve fare, da sempre, un pilota in questi casi?
Deve aumentare a tutti i costi la velocità dell'aeromobile, ed il modo più sicuro per farlo è farlo scendere in modo che acquisti velocità'.
In totale automatismo, come era progettato il sottosistema software MCAS ha fatto questo, e si è quindi instaurata una lotta tra i piloti che cercavano di risalire, non comprendendo da

cosa era dovuta la pericolosa ed errata manovra, e MCAS che cercava di scendere per salvare l'aeromobile.

Ha vinto il software. Anzi, stravinto.

Ora, la questione di fondo è se i piloti conoscessero davvero l'esistenza dell'MCAS, ed in particolare il fatto che era attivo anche durante il volo manuale e che, se necessario, doveva essere disattivato a parte.
Sicuramente, i fatti lo hanno ahimè dimostrato, l'interfaccia utente del software non lo ha rendeva abbastanza evidente, e la documentazione non lo ha evidenziato in maniera sufficiente.

E così si scopre che due tragedie non sono state dovute al malfunzionamento di un sensore o del software dell'aeromobile nel suo complesso, ma a scelte precise, apparentemente ragionevoli ma errate, della progettazione del software di bordo e della documentazione per i piloti.

A questo punto, a coloro che specificatamente si occupano di sviluppo ed integrazione di software di grandi dimensioni, vorrei porre due domande:

- le metodologie e le prassi di progettazione e testing (non di sviluppo) sono affidabili in contesti di questa complessità (ordine di grandezza di 100 milioni di linee di codice);

- lo stato dell'arte della progettazione delle interfacce utente di questi software è sufficiente per gestire in maniera "sicura" decisioni gravi ed improvvise, quali la necessità di "strappare di mano" i comandi ai piloti?

In questo caso, verrebbe da concludere, le risposte dovrebbero essere ambedue negative.

Scrivere a Cassandra—Twitter—Mastodon
Videorubrica "Quattro chiacchiere con Cassandra"
Lo Slog (Static Blog) di Cassandra
L'archivio di Cassandra: scuola, formazione e pensiero

By Marco A. L. Calamari on May 21, 2022.

Canonical link

Exported from Medium on August 27, 2025.